1. Skriv ett uttryck som utgår ifrån employees och returnerar en lista av alla element i listan (All)
2. Skriv ett uttryck som returnera en lista över alla employees i bokstavsordning (OrderBy på FirstName)
3. Skriv ett uttryck som returnerar en lista över alla employees i omvänd bokstavsordning (OrderByDesc på FirstName)
4. Skriv ett uttryck som returnerar alla employees utom de tio första (Skip)
5. Skriv ett uttryck som returnerar en lista över alla employees som har ett FirstName (Where)
6. Skriv ett uttryck som returnerar en lista över alla employees som har ett LastName (Where)
7. Skriv ett uttryck som utgår ifrån employees och returnerar en lista med alla element (Select)
8. Skriv ett uttryck som returnerar den första employees i listan (First, FirstOrDefault – vad är skillnaden mellan de båda? Testa dem på en tom lista).
9. Skapa en ny lista av integers och lägg till några heltal i listan – summera talen med ett uttryck (Sum – Kan du hitta någon metod för att returnera medlet för heltalen i listan?)
10. Skapa en ny lista av integers och lägg till 3 st. 1:or och 3 st. 2:or i listan. Skriv ett Linq-uttryck som returnerar en lista med unika element i listan (Distinct)
11. Skriv ett uttryck som returnerar ett heltal för hur många employees listan innehåller (Count).
12. Skriv ett uttryck som returnerar ”true” ifall listan av employees innehåller någon användare som har ett FirstName som har Length == 4 (Any)
13. Skriv ett uttryck som returnerar en lista av employees som har ett FirstName som är 4 karaktärer långt.
14. Skriv ett Linq-uttryck som returnerar de tio första employees i bokstavsordning (OrderBy, Take)
15. Skriv ett Linq-uttryck som returnerar employees 5-7 i listan (Skip, Take)
16. Skriv ett uttryck som returnerar alla employees sorterade i bokstavsordning, där användarna har ett LastName som är längre än 6 karaktärer (Where, OrderBy)
17. Skriv ett uttryck som returnerar den första employees som saknar ett FirstName (OrderBy, FirstOrDefault)
18. Skriv ett uttryck som returnerar den första employees som saknar ett FirstName och LastName (OrderBy, FirstOrDefault)
19. Det finns en property i klassen Employee som heter Number:  
    Skriv ett Linq-uttryck som returnerar den person som har det lägsta nummer
20. Det finns en property i klassen Employee som heter Number:  
    Skriv ett Linq-uttryck som returnerar den person som har det högsta nummer
21. Det finns en property i klassen Employee som heter Number:  
    Skriv ett Linq-uttryck som summerar alla nummer för alla employees

1. Write an expression that assumes employees and returns a list of all elements in the list (All)

2. Write an expression that returns a list of all employees in alphabetical order (OrderBy First Name)

3. Write an expression that returns a list of all employees in reverse alphabetical order (OrderByDesc First Name)

4. Write an expression that returns all employees except the first ten (Skip)

5. Write an expression that returns a list of all employees who have a FirstName (Somewhere)

6. Enter an expression that returns a list of all employees who have a Last Name (Somewhere)

7. Write an expression that assumes employees and returns a list of all elements (Select)

8. Write an expression that returns the first employees of the list (First, FirstOrDefault - what is the difference between the two? Try them on an empty list).

9. Create a new list of integers and add some integers in the list - to sum up figures with an expression (Sum - Can you find any method to return the agent of the integers in the list?)

10. Create a new list of integers and add 3 pieces. 1's and 3. 2's in the list. Write a Linq expression that returns a list of unique elements in the list (Distinct)

11. Write an expression that returns an integer value for how many employees list contains (Count).

12. Write an expression that returns "true" if the list of employees includes any user who has a FirstName as the Length == 4 (Any)

13. Write an expression that returns a list of employees who have a FirstName is 4 characters long.

14. Write a Linq expression that returns the first ten employees in alphabetical order (OrderBy, Take)

15. Write a Linq expression that returns employees 5-7 in the list (Skip, Take)

16. Write an expression that returns all employees sorted in alphabetical order, where users have a LastName that is longer than 6 characters (Where, OrderBy)

17. Write an expression that returns the first employees that do not have a First Name (OrderBy, FirstOrDefault)

18. Write an expression that returns the first employees that do not have a First Name and Last Name (OrderBy, FirstOrDefault)

19. There is a property of the class named Employee Number:

Write a Linq expression that returns the person who has the lowest number

20. There is a property of the class named Employee Number:

Write a Linq expression that returns the person who has the highest number

21. There is a property of the class named Employee Number:

Write a Linq expression that sums up all the numbers for all employees